Self Adaptive Safe Provisioning of Wireless Power using DCOPs

Cornelis Jan van Leeuwen
TNO
Groningen, The Netherlands
Email: coen.vanleeuwen@tno.nl

Kasım Sinan Yıldırım
Delft University of Technology
Delft, The Netherlands
Email: k.s.yildirim@tudelft.nl

Przemysław Pawełczak
Delft University of Technology
Delft, The Netherlands
Email: p.pawelczak@tudelft.nl

Abstract*—Wireless Power Transfer (WPT) technologies aim at getting rid of cables used by consumer devices for energy provision. As long distance WPT is becoming mature, the health impact of WPT becomes increasingly important to consider. In this paper we look at how to maximize the wireless power transfer to remote devices, while maintaining a safe level of electromagnetic radiation (EMR) for humans that are in the vicinity of the energy transmitters. Classically, this problem can be described as a centralized optimization problem of finding the optimal set of safe power levels at locations of human presence. Instead, we advocate to formulate this problem as an agent-based Distributed Constraint Optimization Problem (DCOP). As a solution to this problem we introduce CoCoA_WPT, a variant of the DCOP solver CoCoA. CoCoA_WPT provides a solution of similar quality to centralized solver even for a large scale network involving over a thousand nodes. Based on CoCoA_WPT, we propose a self-adaptive charging system: Transferring Energy Safely by Self-Adaptation (TESSA). TESSA keeps the charging network safe even when it is perturbed by environmental dynamics. We show that TESSA can reach on average up to 85% of the theoretical optimal maximum total transmitted power (calculated using centralized solution) while satisfying the EMR safety constraints.

I. Introduction

Today's ecosystem of the Internet of Things (IoT) is composed of millions of embedded devices that can monitor and control the physical world [1]. These devices are equipped with several hardware components such as sensors (for sensing), actuators (for actuation), microcontrollers (for computation) and transceivers (for communication), that still consume considerable amount of power. Fortunately, the research efforts on electronic circuits have already decreased the power consumption of these hardware components to a few microwatts [2]. This allowed the provision of power wirelessly by means of harvesting the energy of radio frequency (RF) waves [3], [4]. As the efficiency of harvesting circuits improve, many devices are being powered wirelessly using only RF energy without any external power source such as batteries, e.g. WISP [5].

For sustainable/continuous operation, a wirelessly-powered IoT system requires several dedicated RF energy transmitters that can control their power level to charge nearby receivers *collaboratively*, forming a *Wireless Power Transfer Network (WPTN)* [6], [7], [8]. The deployment of this conceptual network is an important issue for the provision of wireless energy in an efficient way. Since the received power by a receiver is inversely proportional with its squared distance to a particular RF transmitter, finding an optimal strategy to minimize the charging time of the whole set of receivers and energy outage of the transmitters is not a trivial problem [9].

Electromagnetic Radiation Safety: The electromagnetic radiation (EMR) at a particular point can be modeled as a linear function of the received power [10, Sec. III-A], [11, Table 1]. Since, several energy transmitters can be active simultaneously to charge the nearby receivers, it might be the case that the total EMR values at some particular points in the charging area—which have a contribution from all active transmitters—can exceed the limit defined by the RF exposure regulations [12], [10]. Therefore, a *safe-charging* WPTN must ensure that it does not create harmful electromagnetic radiation [13], as it is trying to minimize the charging time by increasing the power levels of the transmitters [12].

The Problem Statement: The maximization of the *total transmitted power* by finding individual transmission power levels of the transmitters, while satisfying the *safety constraints* are two orthogonal optimization objectives in a dynamic charging environment. New energy transmitters, as well as receivers, can be introduced to the charging system. This means that EMR values at some particular locations can exceed the threshold if these new transmitters start provisioning power. Moreover, it is difficult to model and estimate exactly the received power, and in turn the EMR value, due to the environmental dynamics of the RF wave propagation. Therefore, a one-shot offline centralized *charging algorithm* that provides the optimal solution to the aforementioned optimization problem is not feasible.

Self-Adaptive and Safe Wireless Charging: Apparently, *self-adaptivity* shall become a necessary property of a WPT system such that the charging algorithm should transfer the network into a safe state, when the safety constraints are violated. Unfortunately, we are unaware of a *self-adaptive and safe* charging algorithm in the current literature that (i) allows energy transmitters to interact with the energy receivers and sensors locally in a distributed fashion; (ii) does not use a centralized entity to find sub-optimal power levels subject to safety regulations; and that (iii) keeps the network safe via simple interactions among the nodes even though it is perturbed by environmental dynamics.

Contributions: In this paper we propose a method based on Distributed Constraint Optimization (DCOP) solvers to find near-optimal solutions for the optimization problems without any centralized entity and by only using local interactions among the nodes. Accordingly in this paper, we introduce a new wireless charging system called TESSA (Transferring Energy Safely by Self-Adaptation). TESSA is based on a variation of the recently proposed DCOP solver CoCoA [14], and is self-adaptive, in the sense that it runs an algorithm on the transmitter nodes that will find an optimal transmission power levels. This not only keeps a WPTN optimal in terms of power transfer, but also safe with respect to EMR regulations. Within this context, we provide the following contributions to the state of the art:

1) We formulate a distributed constraint optimization problem where the energy transmitter devices maximize the total transmitted power to the receiver devices, i.e. minimize the *charging time*, while keeping the network *safe* in the EMR sense using the measurements from the locally deployed sensors;
2) A variation on the CoCoA algorithm is proposed, called CoCoA_WPT, which enables solving the distributed optimization problem, while making sure the EMR thresholds are never violated;
3) We present a novel self adaptive charging system, TESSA, based on the CoCoA_WPT solver. TESSA transfers the charging network to a safe state, even when perturbed by environmental dynamics such as new energy transmitters joining in the network;
4) We compare the CoCoA_WPT algorithm with the existing DCOP solvers via simulations. Our results show that CoCoA_WPT outperforms existing solvers by finding a solution near the theoretical optimal, reaching on average 85% of the maximum (optimal) solution, in terms of power transmission;
5) Our charging system based on DCOP solvers maintains safe EMR levels, even under reasonable levels of model prediction error, which a centralized solution cannot.

II. Related Work

We provide a brief overview of the related work on wireless power transfer and distributed constraint optimization, which our work builds upon.

A. Wireless Power Transfer

The number of IoT nodes continues to grow exponentially [15], [16]. This exposes a problem of sustainable energy provision to such a mass of (battery-powered) IoT devices. Fortunately, the recent advancements in RF energy harvesting and low-power electronics, e.g. [17], make it feasible to power ultra low-power microcontroller-based IoT devices wirelessly using electromagnetic energy [18]. Wireless power transfer revealed several optimization problems that gained considerable attention from the research community, e.g. the optimization of the harvested power [19], energy outage [20] and charging delay [9].

However, the electromagnetic safety issues provide additional constraints on the wirelessly supplied power. Specifically, since an EMR value that is above the limits of well-defined regulations [21] is considered as a threat to human health [22], it introduces an important constraint to the objective of aforementioned optimization problems.

To the best of our knowledge, there are only two recent studies (from the same authors) that target the optimal wireless power transfer incorporating the human health effects. A one-shot *centralized* solution that maximizes the total transmitted power subject to the safety constraints at each point on a pre-defined deployment area is presented in [10]—in [23], the same problem is handled in a *distributed* fashion. Unfortunately, these two studies have a fundamental drawback: they use deterministic models to estimate EMR. However, RF propagation is non-deterministic and modeling errors might lead to violate the safety constraints. Moreover, the distributed algorithm in [23] is quite complex and composed of several phases. The algorithm employs a distributed redundant constraint reduction algorithm, splits the deployment area into small squares and employs linear programming (LP) by considering the local constraints inside each square. Therefore, it cannot be considered as a self-organizing solution since optimization is not performed by local interactions solely, rather than using the global information within each square.

B. Distributed Constraint Optimization

Distributed Constraint Optimization Problems (DCOPs) are a type of problems from the field of multi-agent systems where agents need to cooperatively assign a set of variables in order to optimize some cost function. DCOPs are an extension of (distributed) Constraint Satisfaction Problems [24], [25], but in DCOPs variables can take values from a finite discrete domain, constraint costs can be any real number, and the goal is to optimize the sum of all constraint costs. In Section IV-A the problem will be formally described.

DCOP solvers can be divided in several categories [26]. The first categorization of solvers divides them into complete and incomplete, which provide either the global optimal solution, or a near-optimal solution near the overall best, respectively. However, since the optimization problems are NP-hard [27], optimal solvers are by definition exponentially slow for increasing problem sizes. Therefore we are more interested in incomplete solvers that are not guaranteed to find the optimal solution, and find a good solution in a feasible time.

1) Iterative solvers: Most incomplete DCOP methods use an iterative approach and belong to the class of local search algorithms. This means that these solvers start with an initial variable assignment and iteratively search the local problem space for a solution that incrementally improves, until it converges to a local optimum. Some examples of solvers belonging to this class are the Distributed Stochastic search Algorithm (DSA) [28], the MGM (Maximum Gain Message) or MGM-2 [29] algorithms.

ACLS and MCS-MGM [30] are local search algorithms that are specially designed to solve a particular type of problems, namely Asymmetric DCOPs (ADCOPs). In ADCOPs constraints yield different costs to their involved agents, and hence the algorithms must take into account the effect an assignment has on other agents as well as the effect it has on itself. This property is especially appropriate in use cases where agents' actions affect their neighbors' performance, and where locally positive choices may actually deteriorate the global performance.

Another iterative algorithm is the Max-Sum algorithm [31]; it is also an incomplete solver, but it does not do local search. Instead the Max-Sum algorithm propagates information through the problem graph by representing it as a factor graph, which is a bipartite graph representing both the agents and the constraints as nodes. The Max-Sum algorithm is known for finding high quality solutions, even when constraints are not binary, but involve multiple variables. Extensions of the Max-Sum algorithm have also allowed it to deal with cyclic graphs [32] and with asymmetric problems [33].

2) Non-iterative solvers: CoCoA: There exist some *complete* non-iterative solvers such as DPOP [34], ADOPT [35] or AFB [36]. However, since these complete solvers are very time consuming, for the WPT problem an incomplete solver is preferable. To the best of our knowledge there is only one non-iterative incomplete approach, which is offered by the CoCoA algorithm [14]. We shall provide a more detailed discussion of the CoCoA algorithm in Section V-B together with our proposed extension: CoCoA_WPT.

The advantage of the non-iterative solvers is that they will immediately present their final solution. This means that *if* the solver is capable of finding a solution that satisfies the EMR constraints, it will immediately find this solution, thus *never* violating the EMR thresholds. This is in contrast to iterative solvers, that might initially violate the constraints, even if it may eventually find a solution that yields more transmitted power.

III. System Model: The Network of Energy Provision

We abstract a WPTN as a graph in the 2D plane, that has different types of nodes representing either energy transmitters (ETs), energy receivers (ERs) or sensor nodes. It is assumed that each ET node is equipped with an antenna that can emit RF waves to charge ER nodes inside its *power transmission range* wirelessly. Besides, each ER node is assumed to be equipped with a RF-harvester circuit that accumulates the harvested energy into the storage component. Moreover, each sensor nodes is assumed to be able to measure the EMR value at its specific location.

We further assume that each ET, ER and sensor node is equipped with a transceiver that allows communication with the other nodes inside their *communication range*. For the sake of simplicity, it is assumed that the power transmission range of the ET nodes are identical to their communication range, and ET, ER and sensor nodes are *stationary* during the optimization process.

A. ET Model

The set of ET nodes is denoted by $T = \{T_1, T_2, \dots, T_n\}$, where n represents the number of ETs. We denote the transmission power of T_i such that $i \in \{1, \dots, n\}$ by P_i and assume that each ET node can modify it by assigning values in the interval $[P_{\min}, P_{\max}]$. The set of ER and sensor nodes with which T_i can communicate and transfer power, i.e. the set of neighbors of T_i, is denoted by $\mathcal{M}_i$.

B. ER Model

We denote the set of ER nodes by $R = \{R_1, R_2, \dots, R_m\}$, where m represents the number of ERs. Each $R_j \in \mathcal{M}_i$ such that $j \in \{1, \dots, m\}$ receives power from the ET node T_i. We model the harvested power at R_j from T_i based on Friis transmission equation as

$$P_{i \to j} \quad = \eta \frac{\gamma}{(d_{ij}+\beta)^2} P_i \tag{1}$$

where γ and β are constants that capture the antenna gains, the wavelength and the environmental properties of the radio wave propagation, d_{ij} denotes the distance between T_i and R_j, and η represents the efficiency coefficient of the RF harvester. We denote the total received power at receiver R_j as

$$\theta_j \quad = \textstyle\sum_{i: R_j \in \mathcal{M}_i} P_{i \to j}. \tag{2}$$

C. Sensor Model

The set of sensor nodes is denoted by $S = \{S_1, S_2, \dots, S_l\}$, where l represents the number of sensors. Each sensor S_k such that $k \in \{1, \dots, l\}$ is able to measure the EMR value at its specific location. We model the EMR value at sensor S_k as a *linear* function of the total transmitted power, as

$$E_k \quad = \rho \textstyle\sum_{i: S_k \in \mathcal{M}_i} \frac{\gamma}{(d_{ik}+\beta)^2} P_i \tag{3}$$

where ρ is a constant that captures the linear relationship between the EMR and received power, and d_{ik} denotes the distance between T_i and S_k.

IV. Problem Description

We define the centralized linear programming problem. Define EMR threshold as α and let $\mathcal{P} = [P_1, \dots, P_n]^T$. We formulate the optimization problem as

$$\begin{aligned} \max_{\mathcal{P}} \quad & \textstyle\sum_j \theta_j \\ s.t. \quad & \forall k : E_k \leq \alpha, \\ & P_{\min}\mathbf{1} \leq \mathcal{P} \leq P_{\max}\mathbf{1} \end{aligned} \tag{4}$$

where $\mathbf{1}$ denotes a vector with all components equal to 1.

A. Translation into a DCOP

In DCOPs problems are described as a tuple $\mathcal{T} = \langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{R} \rangle$. We can formulate the wireless power transfer problem by stating the that every transmitter in T is represented by an agent $\mathcal{A} = \{A_1, A_2, \dots, A_n\}$ and their corresponding transmission powers in P are the variables that are being optimized $\mathcal{X} = \{X_1, X_2, \dots, X_n\}$. One of the main assumptions in DCOP theory is that all variables must have finite discrete domains $\mathcal{D}$, which should contain every possible power level. This means that the interval $[P_{\min}, P_{\max}]$ has to be discretized to contain a finite set of possible values, yielding n variable domains $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$, this discretization can follow a specific set of possible power values that the energy transmitter allows.

Finally in a DCOP, $\mathcal{R}$ is the set of constraints, which map the assignment of variables to a non-negative cost: C: $D_{i_1} \times D_{i_2} \times \ldots \times D_{i_k} \to \mathbb{R}$. In the wireless power transfer problem statement there are two types of constraints. First we have the constraints that represent the ER nodes R with a constraint cost function such that

$$C_{R_j} = -\theta_j. \tag{5}$$

Secondly there is the set of sensors S that are modeled using a threshold constraint function such that

$$C_{S_k} = \begin{cases} 0, & \text{if } E_k < \alpha, \\ \tau, & \text{otherwise,} \end{cases} \tag{6}$$

where a hard constraint can be simulated by choosing $\tau = \infty$ as described in [37], or simply a very high value such that $\theta_j \ll \tau$. Hence the full set of constraints can be defined as the set of both receiver and sensor constraints $\mathcal{R} = \{C_R, C_S\}$ such that the minimization function will maximize the transmitted power, while minimizing the number of sensors where the EMR threshold is violated. The DCOP minimization function is defined simply as

$$\arg\min_{\mathcal{X}} \sum \mathcal{R}. \tag{7}$$

V. TESSA: A Self-Adaptive Wireless Charging System

Having presented the system model of wireless power transfer we consider in this paper, we are ready to present our charging system. We call it *Transferring Energy Safely by Self-Adaptation* (TESSA). TESSA finds maximum wireless energy provided to the energy receivers subject to the EMR constraints. The EMR values required as an input to TESSA optimizer are measured by sensors deployed at specific points in the charging area.

We first present the high-level charging protocol that governs the charging requests of the energy receiver devices. Then, we present the wireless power provision algorithm (denoted as CoCoA_WPT), triggered by TESSA. CoCoA_WPT algorithm description is preceded by the overview of the CoCoA solver, introduced first in [14], on which CoCoA_WPT is based.

A. The Main Charging Protocol

The main charging protocol, executed by each transmitter T_i in the wireless power transfer network, is presented in Algorithm 1. The objective of this protocol is to trigger the CoCoA_WPT algorithm, presented in Algorithm 2 and discussed in Section V-D, that will determine the sub-optimal power levels for the transmitters. Initially, the power transmitter can be turned off and waiting for a receiver inside its neighborhood $\mathcal{M}_i$ to send a **Charge** message. The transmitter will add the corresponding receiver to its requests list R_i and reset

Algorithm 1 Charging Protocol executed by Transmitter T_i

1: $\mathsf{R}_i \leftarrow \emptyset$ ▷ Vector of charge requests for transmitter i

Upon receive **Charge** from $R_j \in \mathcal{M}_i$

2: $\mathsf{R}_i \leftarrow \mathsf{R}_i \cup \{R_j\}$ ▷ Add new request from receiver j

3: **reset** CoCoA_WPT ▷ Execute CoCoA_WPT optimizer

Upon receive **EndCharge** from $R_j \in \mathcal{M}_i$

4: $\mathsf{R}_i \leftarrow \mathsf{R}_i \setminus \{R_j\}$ ▷ Remove request from receiver j

5: **if** $\mathsf{R}_i = \emptyset$ **then**

6: Turn off charger T_i

7: **end if**

8: **reset** CoCoA_WPT ▷ Execute CoCoA_WPT optimizer

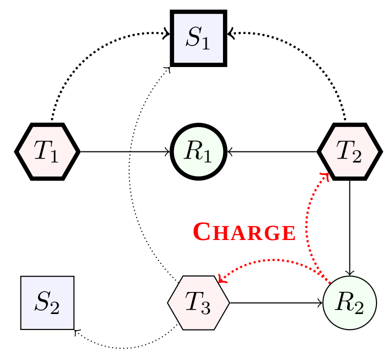

Fig. 1. Illustrative example of TESSA execution. Transmitters T_1 and T_2 are currently charging the receiver R_1 subject to the constraint on the sensor S_1. The receiver R_2 sends a charging request to transmitters T_1 and T_2 that forces T_3, T_2, and in turn T_1, to run the CoCoA_WPT optimizer again with also considering the constraints on the sensor S_2.

the CoCoA_WPT algorithm to recalculate the optimal power levels (Lines 1–3). Similarly, when a **EndCharge** message is received, the corresponding receiver is removed from the requests list, the transmitter is turned off if the requests list is empty and CoCoA_WPT algorithm is restarted (Lines 5–10).

The rationale behind resetting and in turn restarting the actual DCOP solver CoCoA_WPT is to force the network to adapt to the new state. As an example, consider Figure 1: transmitters T_1 and T_2 are currently charging receiver R_1, and receiver R_2 sends a charging request to transmitters T_2 and T_3. Since the sensor S_1 is already in the neighborhood of T_3 and there is a new sensor S_2 that will be solely affected by the power transmission of T_3, the EMR constraints on these sensors will effect not only the power level of T_3 but the current power level of T_2 and in turn T_1. Therefore, if T_3 starts/resets CoCoA_WPT, the neighboring transmitter T_2 should be informed, that will lead T_2 to inform T_1 so that all transmitters re-calculate the optimal power levels.

B. CoCoA Overview

Before introducing CoCoA_WPT, for the clarity of exposition we shall recapitulate on the operation of CoCoA, presented extensively in [14]. The key strategies of CoCoA are: (i) allowing regional information exchange between agents to estimate the effect of an assignment on the neighboring agents, (ii) delaying choices that may lead to suboptimal solutions until more information becomes available, and (iii) use of state machines to avoid race conditions and deadlocks.

The CoCoA algorithm starts at any random agent A_i, which first will send a message to all of its neighbors, inquiring the effect of any possible assignment for their local cost. This will trigger its neighbors to compute for every possible assignment for X_i what the lowest cost would be for A_j, taking into account the current known state and that new assignment. The resulting cost map is sent back to the inquiring agent who can now find the minimizing value, by taking the sum over all received cost maps.

If no unique minimizer is found and at least one neighboring variable is not yet assigned, the algorithm *holds* and waits until a neighbor has updated its variable before the algorithm is run again. This mechanism makes sure that premature choices are avoided until more information is available. If at some point all agents in a region are *holding* the algorithm relaxes the uniqueness restriction until an assignment can be made. Eventually when an agent assigns a value, the algorithm is activated at neighboring agents.

C. CoCoA and Race Conditions

The CoCoA algorithm by itself already has some properties to avoid race conditions, i.e. situations in which two agents simultaneously make a decision on a variable assignment. However, they are not fully effective, since two agents may become simultaneously activated from different neighbors. Also CoCoA is designed under the assumption that there are only binary constraints in the problem graph, which means race conditions are more likely to occur and may involve more than two agents.

If race conditions would occur in the WPT scenario, this means that two (or more) transmitters simultaneously decide on a power level. The situation may occur as shown in Figure 2, where initially two transmitters (T_1 and T_2) are simultaneously actively running the algorithm. As they both know that a shared sensor (S_1) is not exceeding the EMR threshold, they may decide both to increase their power level. Not taking into account the assignment of the neighbor, it is possible for the two agents to assign two values that actually would violate the EMR constraint.

It is very difficult to avoid race conditions from occurring in a multi-agent system—however, since CoCoA already provides a mechanism to disseminate the current state of the algorithm, we can detect if one has occurred. In the following section we introduce the extension of CoCoA: CoCoA_WPT, where race conditions are recognized, and concurrent assignments avoided.

D. Solving CoCoA Race Condition Issue: CoCoA_WPT

In Algorithm 2 we present the pseudocode of the modified CoCoA_WPT algorithm. Note that in the algorithm we use ϕ_i to denote the state of A_i, which initially is `IDLE`, but can be set to `ACTIVE`, `HOLD` or `DONE`. Also the uniqueness bound υ is initially set to 1. It starts out the same way as

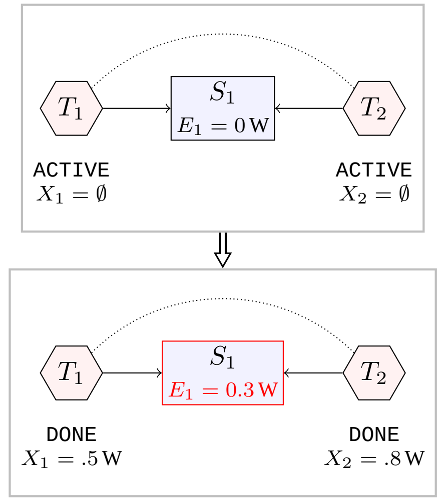

Fig. 2. Race condition problem of CoCoA in the wireless power transfer context. Two transmitters activated at the same time may inadvertently assign power levels that may violate the EMR constraint. In the initial state (top), the transmitters are deciding on a power level. Since S_1 is measuring $0\,\mathrm{W}$ both transmitters decide on a high value eventually exceeding the EMR threshold of $0.18\,\mathrm{W}$ (bottom).

CoCoA does by inquiring the neighbors the costs of local assignments. Then in lines 4–6 the algorithm checks whether any neighboring agents are also currently running, and if there are—it will go back in the algorithm to the point where it will gather information anew. However, since this current agent itself is also running, we would introduce a deadlock here, where two simultaneously activated neighbors would stay in this cycle ad infinitum. In order to break this potential deadlock we introduce this notion of ranking.

In line 4, we check the number of *higher ranked* neighbors. In principle any ranking could be used, as long as all involved agents agree on the ranking. In our implementation we use the alphabetical ranking of the identifier of the variable, but any other rankings, such as based on the physical MAC addresses of the agents may served as well. Even an random number selected at the time of this impasse would serve, as long as there is always *one* highest ranked agent. Only the highest ranked agent may finish the variable assignment, and the other agent(s) will have to restart the algorithm. By doing so we make sure that no two agents are deciding on an assignment simultaneously.

Between line 7 and 12, where the algorithm assigns a variable based on the neighbors' cost messages, the logic is the same as for CoCoA. All reveived costs are added, and the minimizing value is selected if the *uniqueness* of the minimizer is less or equal than υ.

When a message arrives inquiring about the assignment costs (lines 12–18), nodes gather information from the receivers based on either their actual measurements or estimations based on the theoretical energy harvesting model as per (2), and from the sensors by requesting their measured power level. By using the actual measured power, not the

Algorithm 2 CoCoA_WPT Algorithm

```
    Algorithm start on A_i:
 1: Assign φ_i ← ACTIVE and inform neighbors
 2: Send request to neighbors for cost maps
 3: Wait for all responses
 4: if number of ACTIVE higher ranked neighbors > 0 then
 5:     Go to line 2
 6: end if
 7: Find minimizing assignments for X_i
 8: if number of minimizer ≤ υ or number of idle/active
    neighbors is 0 then
 9:     Assign X_i and φ_i ← DONE; send to all neighbors
10: else
11:     Assign φ_i ← HOLD and send to all neighbors
12: end if

    Upon receiving cost inquiry message at A_j:
13: Get θ_r from for every ∀R_r ∈ M_j
14: Get measurements E_s for every ∀S_s ∈ M_j
15: for all X_{i,k} ∈ D_i do
16:     Calculate costs for X_{i,k} ∩ θ_r ∩ E_s
17: end for
18: Reply A_i with all costs

    Upon receiving new state φ_i from A_i ∈ M_j on A_j:
19: Store neighbor's state φ_i
20: if φ_i is HOLD and φ_j = HOLD and number of idle/active
    neighbors is 0 then
21:     Increment uniqueness bound υ and repeat algorithm
22: else if φ_i is DONE and φ_j is HOLD then
23:     Repeat algorithm
24: end if

    Upon receiving RESET on A_i:
25: if φ_j ≠ IDLE then
26:     Assign φ_j ← IDLE
27:     Forward RESET to all neighbors
28: end if
```

model predicted amount, we can make sure that the EMR thresholds are always satisfied. Using this information we can compute the local cost with (5) and (6) in line 16.

For handling the state update message from neighbors (lines 19—24) the same logic is applied as in the original CoCoA algorithm. Whenever a `HOLD` state is received, the algorithm checks if the uniqueness bound υ needs to be updated, or it will repeat the algorithm if the agent itself is in the `HOLD` state and a neighbor informs us that it is finished.

Finally an additional message was added to reset the algorithm. At line 25 we specify that if a **RESET** message is received, it will update the local state to `IDLE` and forward the message to the neighboring agents, if the state was not already reset.

VI. Experiments

We performed four experiments to evaluate the performance of the different methods in the wireless power transfer scenario. For all experiments we generate 200 problem instances and let all methods solve the same problem instance. For the DCOP solvers we had to discretize the potential power levels into $|D| = 20$ levels linearly spaced in $[P_{\min}, P_{\max}]$ where $P_{\min} = 0$ and $P_{\max} = 10$. We set the power transmission variables $\gamma = \beta = 100$ and $\eta = \rho = 1$. The EMR threshold was set to $\alpha = 0.018$, and the threshold violation cost $\tau = 10^9$. For all experiments we define the total solution cost as the sum of all constraint costs as per (5) and (6), i.e. negative the amount of total power consumption plus τ times the number of sensors where the EMR threshold is violated.

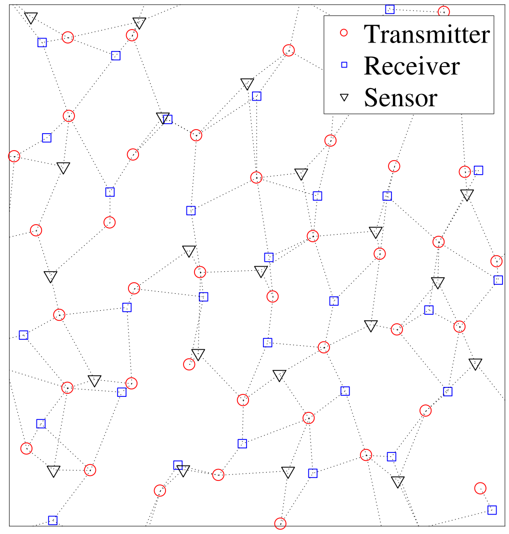

Fig. 3. An example randomized graph as generated using the proposed methods for the simulation experiments with $n = 70$, $m = 60$ and $l = 50$. There are transmitters with 1 up to 4 receivers visible.

To generate the problem graph we selected the position of the n transmitters using a Poisson point process in 2D space. Subsequently the m receiver and l sensor locations are also selected using the same method, and their locations are then scaled such that they span the same area. We determine the average distance of the third-nearest neighbor in Euclidean space and define that any sensor or receiver that is within this distance, is a neighbor of the transmitter, and hence will receive energy. Using this method we can generate a seemingly natural distribution of nodes with some variation in the number of neighbors that a transmitter has. An example of such a generated graph is shown in Figure 3, with $n = 70$, $m = 60$ and $l = 50$.

All experiments were performed in simulation on a laptop with an Intel Core i7-6600U at 2.6 GHz and 16 GB of RAM. For reproducibility of the results all code is available from https://github.com/coenvl/jSAM (solvers and problem definition in java) and https://github.com/coenvl/mSAM (experiments and figures in matlab).

A. Experiment 1: Comparing Solvers

In the first experiment we compare the performance of various DCOP solvers with the centralized solver. For this

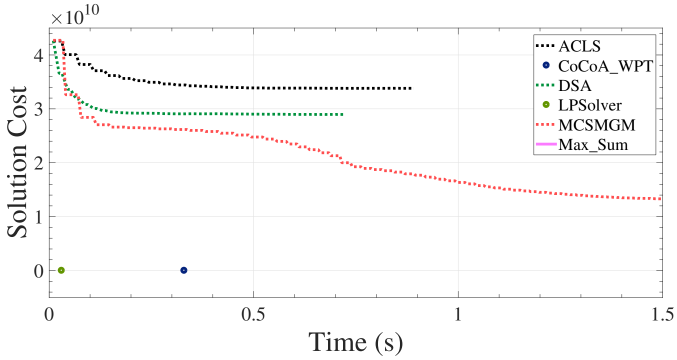

Fig. 4. Various iterative algorithms fail to find a solution that satisfy the receiver constraints. CoCoA_WPT however is capable of finding a valid solution, similar to the centrally computed optimum.

TABLE I
NUMERICAL RESULTS OF THE DCOP SOLVERS

Algorithm	I	S	M	T
CoCoA_WPT	N/A	-1.932	1687	0.3 s
ACLS	27	33×10^9	10378	0.3 s
DSA	16	28×10^9	3148	0.2 s
MCS-MGM	117	13×10^9	34595	1.1 s
Max_Sum	49	4×10^7	30079	321.7 s

experiment we generated problems with $n = 100$, $m = 75$ and $l = 50$. We compare the results of the DCOP solvers ACLS, CoCoA_WPT, DSA, Max-Sum and MCS-MGM with a centralized LP solver.

From the results in Figure 4 we can see that from all DCOP solvers, only CoCoA_WPT is consistently capable of finding solutions that satisfy the EMR threshold constraints. Averaged over 200 experiments, it found a solution that transmits 1.93 W in 0.33 s compared to 2.15 W in 0.03 s for the centralized LP solver. Of course because it is constrained in the exact values of the power levels, its final solution cost is somewhat worse. The Max-Sum algorithm did perform relatively well, but took very long to solve the problem, hence was left out from Figure 4.

In Table I the numeric results of the experiment are shown. For every algorithm the amount of iterations (I), the final solution cost (S), the number of transmitted message (M) and the time to solve (T). The Max-Sum algorithm violated constraints 3.5% of the time, but if it does not, the mean cost is –2.067 W, which is slightly better than CoCoA's mean result. Only in one instance out of the 200 experiments did MCS-MGM find a solution that did not violate the EMR constraints, whereas DSA and ACLS never did.

B. Experiment 2: Scalability

In the second experiment we investigate the performance of the CoCoA_WPT algorithm under varying problem sizes. Fixing all other parameters of the problem we generate instances with increasingly more transmitters (varying between 4 and 1024), and 0.8 times as many receivers and 0.6 times as many sensors.

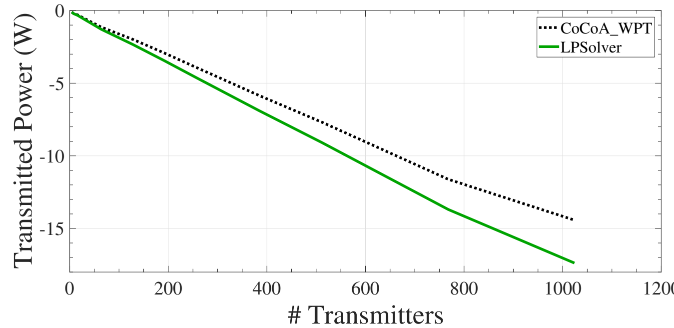

Fig. 5. CoCoA_WPT has the capability of solving even larger graphs, and the distance to the global optimum is linearly dependent on the graph size.

In Figure 5 the solution cost is shown for CoCoA_WPT and the LP solver for increasingly large problems. As can be seen, CoCoA_WPT performance is only linearly worse than the optimal solution for the problem size. On average the solution by CoCoA_WPT yields 85% the amount of power compared with the optimal solution found by the LP solver.

C. Experiment 3: Performance Under Model Error

In the third experiment we validate our hypothesis that using a DCOP approach will keep performing well, even under unpredictable amounts of energy transmitted. In the TESSA charging system, the sensor nodes communicate to the ETs, their actual measurements of the EMR values which do not always perfectly follow the model as proposed in Section III; for example because of quantization effects. Similarly, the ET nodes can either use (i) the amount of measured harvested power based on the measurements by the ERs or (ii) the predicted total harvested power based on the theoretical model represented by in (2).

In all previous experiments we assumed that (i) our theoretical model in Section III perfectly represents the amount of transmitted energy and (ii) both sensors and receivers perform measurements reflected by the equations (2) and (3). In order to explore the effect of the measurements on the performance of our system we introduce an error in the model on how much energy is received by the receivers and the sensors. i.e. for any combination of transmitter i and receiver j the amount of harvested energy is

$$P'_{i \to j} = \epsilon \eta \frac{\gamma}{(d_{ij}+\beta)^2} P_i \tag{8}$$

and similarly the amount of EMR measured by a sensor k is

$$E'_k = \epsilon \rho \sum_{i: S_k \in \mathcal{M}_i} \frac{\gamma}{(d_{ik}+\beta)^2} P_i, \tag{9}$$

where ϵ is a random noise multiplier from the normal distribution $\epsilon \sim \mathcal{N}\left(1, \sigma^2\right)$: white noise added to the amount of transmitted power from the original model.

In Figure 6 the results are shown for the CoCoA_WPT algorithm as it solves different problems with an increasing amount of noise, compared to the centralized LP solver. We observe that our solver performs well by continuously satisfying the EMR constraints. The centralized LP solver

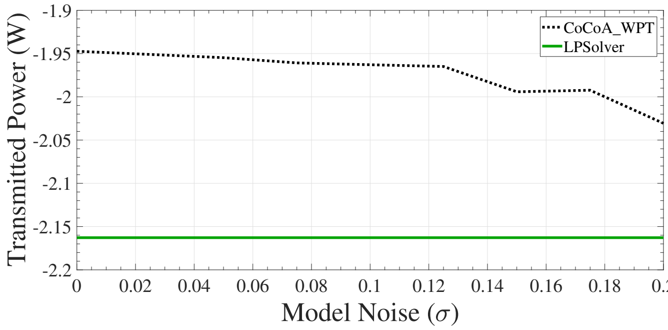

Fig. 6. Noise in the model is not of large influence to the solution quality of the DCOP solver; however the central solver using the same noise would not generate valid solutions. The line in the graph shows the theoretical optimum *without* model noise.

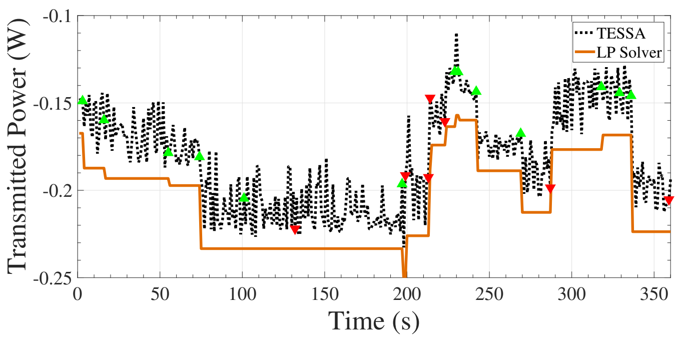

Fig. 7. TESSA correctly reacts to disturbances in the environment. We can see that it finds safe solutions that are always near the optimal solution found by the LP solver. Upward arrows (green) indicate a random transmitter was added to the environment, and downward triangles (red) indicate the removal of a transmitter.

makes its assignments based on the predetermined energy harvesting and EMR model and cannot take into account the actual measurements. Consequently, because of modeling and measurement errors, in practical scenarios it is impossible to estimate the actual harvested power and in turn the EMR values [12]. If we apply the solution found by the LP solver in the noisy model, we find that in all instances except where $\sigma = 0$, some sensor constraints were violated, leading to invalid solutions.

D. Experiment 4: Dynamic Environment

In the final experiment we investigate the performance of the TESSA charging system under network dynamics. In this experiment we generate the a network with 10 transmitters, 8 receivers and 6 sensors. We run the TESSA charging system, and randomly add or remove agents. Specifically in every second there is a 5% chance that the network will change, and if it does, then half of the times a transmitter is added, and half the time a randomly selected transmitter is removed from the WPTN. The CoCoA_WPT algorithm is reset every second.

In Figure 7 the total amount of transmitter power is presented for both TESSA, and for the centralized LP solver that calculates the optimal power levels. Observe that when a transmitter is removed from or added to the charging system which is currently charging receivers, TESSA disseminates a **RESET** message (see Algorithm 1) to start the optimization process again. Therefore, the charging network reacts to this disturbance by re-calculating the optimal power levels of the whole transmitters with respect to the EMR safety constraints. Eventually, all transmitters will start transmitting energy according to the new power levels that comply with the new safety constraints in accordance with the new structure of the network.

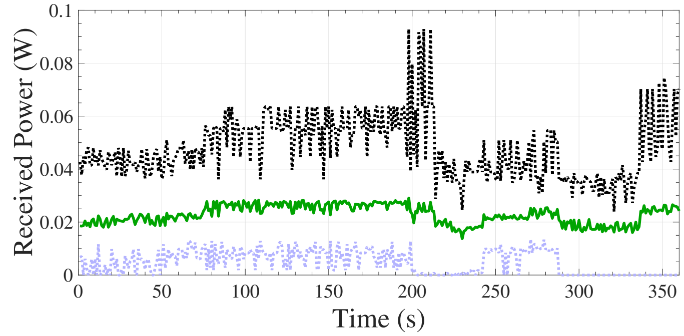

Fig. 8. Example of tracking the amount of received power for all ER in the dynamic experiment, showing the minimum (light blue), average (green) and maximum (black).

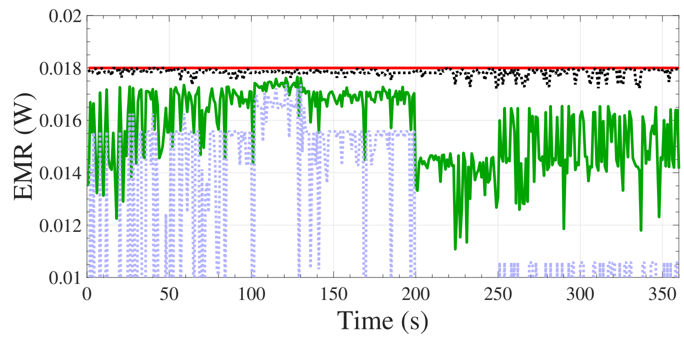

Fig. 9. The EMR is logged for every sensor in the dynamic environment experiment. Here we show the EMR threshold at 0.018 W (red line), together with the minimum (light blue), average (green) and maximum (black) measured EMR of all sensors.

This network adaptivity is almost impossible to achieve with the centralized LP solver. The reason is that, for each transmitter removal/addition, the whole state of the network—including transmitter power levels, the positions of the receivers and the sensors—should be collected and sent to the central entity that calculates the optimal LP solution. What is more, the results of these calculations should be distributed back to the corresponding transmitters so that they update their power levels. Thanks to TESSA and the CoCoA_WPT solver, adaptivity is achieved by only the local interactions among the agents in our system.

VII. Discussion

In this paper we introduced a new charging system TESSA, for safe wireless power transfer, that utilizes an efficient

DCOP solver CoCoA_WPT to ensure that electromagnetic radiation levels never exceed safety guidelines. To this end, we formalized the safe wireless charging problem as a DCOP so that any DCOP solver can be used to solve this problem. Then, we introduced a variant of the CoCoA, namely CoCoA_WPT solver that solves the aforementioned problem in an efficient way in a dynamic network. We compared CoCoA_WPT with the existing solvers and justified that it is capable of consistently finding solutions that maintain safe levels of EMR. Then, we presented experiments that showed the TESSA charging system is self-adaptive in the sense that it reacts to the network dynamics and always transfers the network to an EMR-safe state, meanwhile optimizing the total transmitted power.

Our proposed charging method is based on an extension of the non-iterative CoCoA algorithm that guarantees the exclusion concurrent assignments due to race conditions. We show that it consistently finds solutions that maintain the EMR thresholds, whereas other DCOP solvers could not. The amount of transferred power was on average 85% the amount of power transferred in the theoretical optimal conditions, independent of the problem size. The losses can be partially explained by the fact that the solver is incomplete, so better solutions may be possible. Also, because the DCOP solver can only choose a power level from a finite set, whereas in the optimal setting any power level between the minimum and the maximum can be selected. We also showed that our method was able to find a solution that satisfies the EMR threshold, even when there is a moderate amount of model prediction error in the amount of transferred energy, by communicating measurements from sensors to transmitters.

We did not perform any experiments involving hardware implementations, but this would be the reasonable next step in providing safe wireless charging solutions. This will yield valuable information about how well the methods perform with realistic disturbances and practical problems.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] S. Gollakota, M. Reynolds, J. Smith, and D. Wetherall, "The emergence of RF-powered computing," *IEEE Computer*, vol. 47, no. 1, pp. 32–39, Jan. 2014.

[3] D. Mishra, S. De, S. Jana, S. Basagni, K. Chowdhury, and W. Heinzelman, "Smart RF energy harvesting communications: challenges and opportunities," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 70–78, Apr. 2015.

[4] H. J. Visser and R. J. M. Vullers, "RF energy harvesting and transport for wireless sensor network applications: Principles and requirements," *Proc. IEEE*, vol. 101, no. 6, pp. 1410–1423, Jun. 2013.

[5] (2014) Wisp 5.0 wiki. [Online]. Available: http://wisp5.wikispaces.com

[6] Q. Liu, M. Golińński, P. Pawełczak, and M. Warnier, "Green wireless power transfer networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1740–1756, May 2016.

[7] O. Galinina, K. Mikhaylov, K. Huang, S. Andreev, and Y. Koucheryavy, "Wirelessly powered urban crowd sensing over wearables: Trading energy for data," 2016. [Online]. Available: https://arxiv.org/abs/1611.05910

[8] K. Huang and X. Zhou, "Cutting the last wires for mobile communications by microwave power transfer," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 86–93, Jun. 2015.

[9] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Minimizing charging delay in wireless rechargeable sensor network," in *Proc. INFOCOM*, Turin, Italy, Apr. 14–19, 2013.

[10] H. Dai, Y. Liu, G. Chen, X. Wu, and T. He, "Safe charging for wireless power transfer," in *Proc. INFOCOM*. Toronto, Canada: IEEE, Apr. 27–May 2 2014, pp. 1105–1113.

[11] S. He, J. Chen, F. Jiang, D. K. Yau, G. Xing, and Y. Sun, "Energy provisioning in wireless rechargeable sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 10, pp. 1931–1942, Oct. 2013.

[12] Q. Liu, K. S. Yildirim, P. Pawełczak, and M. Warnier, "Safe and secure wireless power transfer networks: Challenges and opportunities in RF-based systems," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 74–79, Sep. 2016.

[13] T. Wu, T. S. Rappaport, and C. M. Collins, "Safe for generations to come: Considerations of safety for millimeter waves in wireless communications," *IEEE Microw. Mag.*, vol. 16, no. 2, pp. 65–84, Mar. 2015.

[14] C. J. van Leeuwen and P. Pawełczak, "CoCoA: A non-iterative approach to a local search (A)DCOP solver," in *AAAI Conference on Artficial Intelligence*, San Fransisco, CA, USA, Feb. 4–11 2017.

[15] F. K. Shaikh, S. Zeadally, and E. Exposito, "Enabling technologies for green internet of things," *IEEE Syst. J.*, vol. PP, no. 99, pp. 1–12, 2015.

[16] E. Bertino, K.-K. R. Choo, D. Georgakopolous, and S. Nepal, "Internet of things (IoT): Smart and secure service delivery," *ACM Trans. Internet Technol.*, vol. 16, no. 4, pp. 22:1–22:7, Dec. 2016.

[17] H. Gao, M. K. Matters-Kammerer, P. Harpe, D. Milosevic, A. van Roermund, J.-P. Linnartz, and P. G. Baltus, "A 60-GHz energy harvesting module with on-chip antenna and switch for co-integration with ULP radios in 65-nm CMOS with fully wireless mm-wave power transfer measurement," in *Proc. ISCAS*, Melbourne, Australia, Jun. 1–5 2014, pp. 1640–1643.

[18] J. R. Smith, *Wirelessly Powered Sensor Networks and Computational RFID*. New York, NY, USA: Springer Verlag, 2013.

[19] I. Krikidis, "Simultaneous information and energy transfer in large-scale networks with/without relaying," *IEEE Trans. Commun.*, vol. 62, no. 3, pp. 900–912, Mar. 2014.

[20] K. Huang and V. K. Lau, "Enabling wireless power transfer in cellular networks: Architecture, modeling and deployment," *IEEE Trans. Wireless Commun.*, vol. 13, no. 2, pp. 902–912, 2014.

[21] C. Kalialakis and A. Georgiadis, "The regulatory framework for wireless power transfer systems," *Wireless Power Transfer*, vol. 1, no. 2, pp. 108–118, Sep. 2014.

[22] A. Ahlbom, U. Bergqvist, J. Bernhardt, J. Cesarini, M. Grandolfo, M. Hietanen, A. Mckinlay, M. Repacholi, D. Sliney, J. Stolwijk *et al.*, "Guidelines for limiting exposure to time-varying electric, magnetic, and electromagnetic fields (up to 300 GHz)," *Health Phys.*, vol. 74, no. 4, pp. 494–522, Apr. 1998.

[23] H. Dai, Y. Liu, G. Chen, X. Wu, and T. He, "SCAPE: Safe charging with adjustable power," in *Proc. ICDCS*, Madrid, Spain, Jun. 30–Jul. 3 2014, pp. 439–448.

[24] J. Liu and K. P. Sycara, "Exploiting problem structure for distributed constraint optimization," in *Proc. ICMAS*, San Fransisco, CA, USA, Jun. 12–14 1995, pp. 246–253.

[25] M. Yokoo, E. H. Durfee, T. Ishida, and K. Kuwabara, "The distributed constraint satisfaction problem: Formalization and algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 10, no. 5, pp. 673–685, Sept./Oct. 1998.

[26] A. R. Leite, F. Enembreck, and J.-P. A. Barthès, "Distributed constraint optimization problems: Review and perspectives," *Expert Systems with Applications*, vol. 41, no. 11, pp. 5139–5157, Sep. 2014.

[27] P. J. Modi, "Distributed constraint optimization for multiagent systems," Ph.D. dissertation, University of Southern California, Los Angeles, CA, USA, 2003.

[28] W. Zhang, G. Wang, X. Zhao, and L. Wittenburg, "Distributed stochastic search and distributed breakout: properties, comparison and applications to constraint optimization problems in sensor networks," *Artificial Intelligence*, vol. 161, no. 1, pp. 55–87, Jan. 2005.

[29] R. T. Maheswaran, J. P. Pearce, and M. Tambe, "Distributed algorithms for DCOP: A graphical-game-based approach," in *Proc. ICPADS*, San Fransisco, CA, USA, Sep. 15–17 2004, pp. 432–439.

[30] T. Grinshpoun, A. Grubshtein, R. Zivan, A. Netzer, and A. Meisels, "Asymmetric distributed constraint optimization problems," *Journal of Artificial Intelligence Research*, vol. 47, pp. 613–647, May 2013.

[31] A. Farinelli, A. Rogers, A. Petcu, and N. R. Jennings, "Decentralised coordination of low-power embedded devices using the max-sum algorithm," in *Proc. AAMAS*, Estoril, Portugal, May 12–16 2008, pp. 639–646.

[32] R. Zivan and H. Peled, "Max/min-sum distributed constraint optimization through value propagation on an alternating DAG," in *Proc. AAMAS*, Valencia, Spain, Jun. 4–8 2012, pp. 265–272.

[33] R. Zivan, T. Parash, and Y. Naveh, "Applying max-sum to asymmetric distributed constraint optimization," in *Proc. IJCAI*, Buenos Aires, Argentina, Jul. 25–31 2015, pp. 432–438.

[34] A. Petcu and B. Faltings, "A scalable method for multiagent constraint optimization," in *Proc. IJCAI*, Edinburgh, Scotland, UK, July 30–Aug. 5 2005, pp. 266–271.

[35] P. J. Modi, W.-M. Shen, M. Tambe, and M. Yokoo, "ADOPT: asynchronous distributed constraint optimization with quality guarantees," *Artificial Intelligence*, vol. 161, no. 1–2, pp. 149–180, Jan. 2005.

[36] A. Gershman, A. Meisels, and R. Zivan, "Asynchronous forward bounding for distributed COPs," *Journal of Artificial Intelligence Research*, vol. 34, no. 1, pp. 61–88, Jan. 2009.

[37] J. C. Boerkoel and E. H. Durfee, "Distributed reasoning for multiagent simple temporal problems," *Journal of Artificial Intelligence Research*, vol. 47, pp. 95–156, 2013.